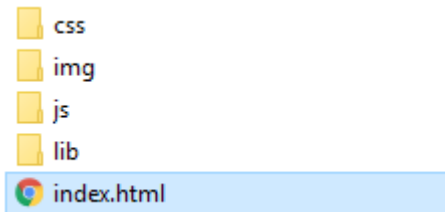


# Module 1

## CSS

### Structure d'un projet Web



- La page principale du site Web se nomme toujours « index.html ».
- Les autres pages du site Web peuvent se retrouver à la racine du projet ou dans un autre dossier.
- On va séparer les fichiers scripts dans le dossier « js », les fichiers de styles dans le dossier « css » et les images dans le dossier « img ».
- Le dossier « lib » sert pour les librairies externes que l'on va utiliser.
- On utilise notre jugement pour l'organisation et la structure du projet.

### Lier la feuille de style à une page Web

```
<link href="style.css" rel="stylesheet" />
```

- On mets cette balise dans le « head » de notre page.
- Si on a plusieurs feuilles de styles, on ajoute simplement plusieurs balise « link ».

### Règle de style

```
div {  
  background-color: #FF0000;  
}
```

- Sélecteur : Le sélecteur permet de définir sur quels éléments nous appliquerons le style.

- Déclaration de style : Une déclaration de style est composée d'un nom de propriété et d'une valeur.

## Sélecteurs

Le sélecteur permet de définir sur quels éléments nous appliquerons le style.

### Sélecteur par balise

On écrit simplement le nom de la balise HTML. Toute les balises HTML de ce type seront affectées.

```
div
```

### Sélecteur par id

Sélectionne les éléments de la page HTML par leur attribut **id**. Un id dans une page Web devrait être unique. Pour indiquer qu'on sélectionne les id, on utilise le symbole « # » devant le id.

```
<div id="unId"></div>
```

```
#unId { ... }
```

### Sélecteur par classe

Sélectionne les éléments de la page par leur attribut **class**. Les classes dans une page Web peuvent être utilisé à plusieurs endroits. Pour indiquer qu'on sélectionne une classe, on utilise le symbole « . » devant la classe.

```
<div class="uneClasse"></div>
```

```
.uneClasse { ... }
```

### Sélecteur complexe

Sélectionne les éléments qui respecte plusieurs sélecteurs. On concatène simplement tous les sélecteurs un à la suite de l'autre. Dans l'exemple suivant, on sélectionne les éléments qui sont un **div**, qui ont l'id « unId » et qui ont la classe « uneClasse ».

```
<div id="unId" class="uneClasse"></div>
```

```
div#unId.uneClasse { ... }
```

## Sélecteur multiple

Permet de sélectionner plusieurs sortes d'éléments. On mets simplement une virgule entre chaque sélecteur.

```
div, #unId, .uneClasse, .uneAutreClasse { ... }
```

## Sélecteur intérieur

Permet de spécifier quels élément nous voulons sélectionner à l'intérieur d'un autre élément. On sépare simplement les deux sélecteurs par un espace. Dans l'exemple suivant, on sélectionne tous les éléments ayant la classe « uneClasse » et qui sont dans des **div**, peu importe la profondeur à auxquels ils se trouvent.

```
div .uneClasse { ... }
```

## Autres sélecteurs

Il existe de nombreux autres sélecteurs. Vous pouvez les trouver sur la page Web suivante :

[https://www.w3schools.com/cssref/css\\_selectors.asp](https://www.w3schools.com/cssref/css_selectors.asp)

## Déclaration de style

Il y a une quantité phénoménale de propriétés en CSS. Il y en a tellement que nous ne pourrions pas tous les couvrir dans ce cours. Une petite recherche sur Internet pourra souvent résoudre vos problèmes.

## Hauteur et largeur

```
width: 80%;  
height: 200px;
```

Permet de spécifier la hauteur et la largeur des éléments en pixel ou en pourcentage. Les hauteurs en pourcentage nécessite que la hauteur du parent soit spécifié explicitement dans une autre règle CSS.

## Display

```
display: block;  
display: inline;  
display: none;
```

Il existe de nombreuses valeurs possibles pour la valeur de **display**, mais on verra souvent apparaître soit **block**, **inline** ou **none**.

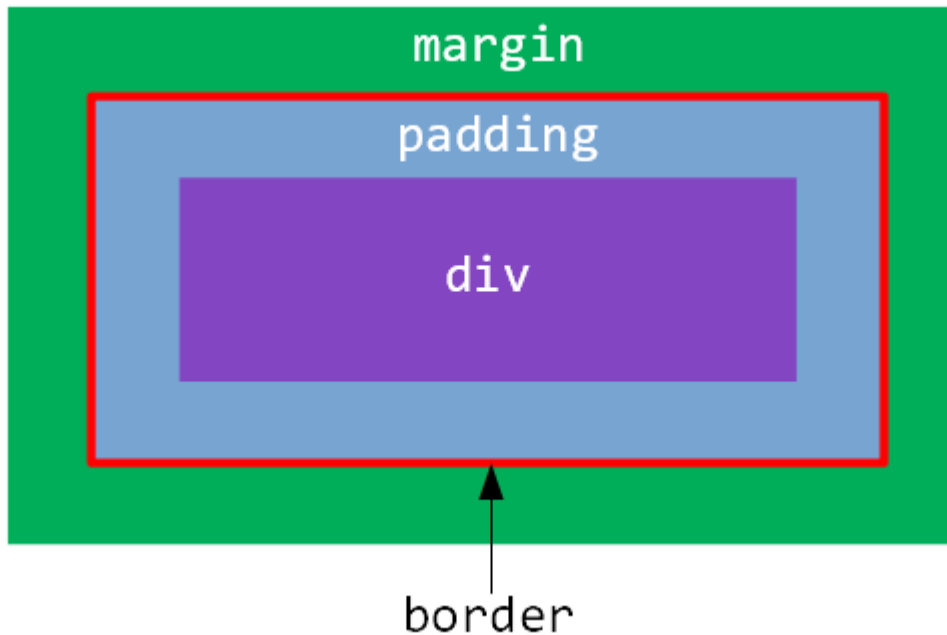
- Block : Affiche l'élément comme s'il était une boîte (un div).
- Inline : Affiche l'élément comme s'il était du texte ou un span.
- None : N'affiche simplement pas l'élément. L'élément est complètement retiré de la page.

Pour plus d'information, visitez le site Web suivant :

- [https://www.w3schools.com/cssref/pr\\_class\\_display.asp](https://www.w3schools.com/cssref/pr_class_display.asp)

## Padding et margin

```
padding: 20px;  
margin: 15px;
```



- padding : Permet de spécifier la quantité d'espace vide autour de l'élément, mais à l'intérieur de la bordure.
- margin : Permet de spécifier la quantité d'espace vide autour de l'élément à l'extérieur de la bordure.

Il est possible de spécifier le margin et padding pour chaque côté de l'élément séparément. Pour plus d'information, vous pouvez vous rendre sur les sites Web suivant :

- [https://www.w3schools.com/css/css\\_boxmodel.asp](https://www.w3schools.com/css/css_boxmodel.asp)
- [https://www.w3schools.com/css/css\\_margin.asp](https://www.w3schools.com/css/css_margin.asp)
- [https://www.w3schools.com/css/css\\_padding.asp](https://www.w3schools.com/css/css_padding.asp)

## Background

```
background-color: transparent;
background-color: red;
background-color: #FF0000;
background-color: #F00;
background-color: rgb(255, 0, 0);
background-color: rgba(255, 0, 0, 0.7);

background-image: url("../img/image.jpg");
background-attachment: scroll;
background-repeat: repeat;
background-position: left top;

background: #FF0000 url(..img/image.jpg) repeat scroll left top;
```

- background-color : Permet de spécifier la couleur de fond de l'élément.
- background-image : Permet de spécifier une image de fond pour l'élément.
- background-attachement : Spécifie si l'image de fond bouge avec la page ou non.
- background-repeat : Indique si l'image de fond doit se répéter ou non.
- background-position : Indique la position de départ de l'image de fond.
- background : Version courte pour spécifier un fond.

Vous pouvez utiliser le lien suivant pour plus d'information sur les différentes valeurs possibles pour chaque propriété pour le **background** :

- [https://www.w3schools.com/css/css\\_background.asp](https://www.w3schools.com/css/css_background.asp)

Il existe d'autres propriétés pour le **background** qui sont disponible depuis CSS3. On peut trouver quelques exemples aux liens suivant :

- [https://www.w3schools.com/css/css3\\_backgrounds.asp](https://www.w3schools.com/css/css3_backgrounds.asp)
- [https://www.w3schools.com/css/css3\\_gradients.asp](https://www.w3schools.com/css/css3_gradients.asp)

## Listes

```
list-style-type: circle;  
list-style-type: lower-alpha;  
  
list-style-image: url("../img/image.png");  
  
list-style-position: inside;  
  
list-style: circle inside url("../img/image.png");
```

- list-style-type : Spécifie le type de puce ou de numérotation à utiliser.
- list-style-image : Permet de spécifier une image comme puce à utiliser.
- list-style-position : Indique si les puces ou la numérotation doivent être dans le « flow » de la page ou non.
- list-style : Version courte pour mettre un style sur des listes.

Pour plus d'information, vous pouvez vous rendre sur le site Web suivant :

- [https://www.w3schools.com/css/css\\_list.asp](https://www.w3schools.com/css/css_list.asp)

## Bordures

```
border-width: 2px;  
  
border-style: solid;  
  
border-color: red;  
  
border: 2px solid black;
```

- border-width : Spécifie la largeur de la bordure.
- border-style : Spécifie le type de ligne à utiliser pour la bordure (ligne solide, avec des espaces, etc.).
- border-color : Spécifie la couleur de la bordure.
- border : Version courte pour mettre un style sur une bordure.

Il est possible de mettre des styles sur des bordures spécifiques d'un élément (bordure du haut, droite, gauche et bas). Pour plus d'information, vous pouvez vous rendre sur le site Web suivant :

- [https://www.w3schools.com/css/css\\_border.asp](https://www.w3schools.com/css/css_border.asp)

## Polices

```
font-family: 'Open Sans', sans-serif;  
font-style: italic;  
font-size: 1.25em;  
font-weight: bold;  
font-variant: small-caps;
```

- font-family : Spécifie la famille de police à utiliser. On peut en mettre plusieurs comme « fallback » si les polices ne sont pas disponible sur l'ordinateur de l'utilisateur.
- font-style : Spécifie si la police doit être en italique ou non.
- font-size : Spécifie la taille de la police. On peut utiliser le pixel, le pourcentage ou le em comme taille. On utilisera généralement le em puisqu'il peut être changé de taille par l'utilisateur assez facilement.
- font-weight : Spécifie si la police doit être en gras.
- font-variant : Spécifie si nous utilisons la variante « small-cap » (Tous les caractères en majuscule, mais plus petit pour quand même différencier les lettres majuscules des lettres minuscules).

Toutes les polices ne supportent pas nécessairement tous les paramètres. Certaines polices ne supporte pas le « small-cap », d'autres ont plusieurs degré de « bold » (en gras). Pour plus d'information visitez les liens suivant :

- [https://www.w3schools.com/css/css\\_font.asp](https://www.w3schools.com/css/css_font.asp)
- <https://fonts.google.com/>
- <https://developer.mozilla.org/en-US/docs/Web/CSS/font-family>

## Textes

```
color: black;
text-align: center;
text-decoration: underline;
text-transform: uppercase;
text-indent: 20px;
line-height: 1.5;
```

- color : Spécifie la couleur du texte.
- text-align : Spécifie l'alignement du texte à l'intérieur de son conteneur.
- text-decoration : Spécifie une décoration pour le texte (souligné, barré, etc.)
- text-transform : Spécifie si on doit mettre tout en minuscule, majuscule, ou seulement mettre la première lettre en majuscule.
- text-indent : Spécifie l'indentation du texte sur la première ligne d'un paragraphe.
- line-height : Spécifie la hauteur des lignes de texte. Ça ne change pas la hauteur de la police, mais ça nous permet de faire des choses comme du double interligne.

Il existe beaucoup d'autres propriétés pour le texte. Vous pouvez en trouver d'autre si vous vous rendez à ce lien :

- [https://www.w3schools.com/css/css\\_text.asp](https://www.w3schools.com/css/css_text.asp)

## Float et positionnement

```
float: left;
position: relative;
```

- float : Permet à un élément d'échapper au flux normal de la page à gauche ou à droite. On l'utilise beaucoup pour les pages Web « responsive » (qui change de disposition sur les appareils mobiles). Cette propriété cause souvent beaucoup de problèmes et est difficile à gérer, mais elle est très utile, surtout pour les vieux navigateurs. On utilise souvent un truc appelé Clearfix pour s'assurer que les éléments qui flottent ne détruisent pas la disposition du



reste de la page. Pour plus d'information sur la propriété **float**, visitez le site Web suivant :

[https://www.w3schools.com/css/css\\_float.asp](https://www.w3schools.com/css/css_float.asp)

- position : Permet de spécifier comment les éléments sont placés dans la page. La gestion du positionnement est assez complexe, mais permet placer les éléments aux bonnes place dans votre page Web. Pour plus d'information sur la propriété **position**, visitez le site Web suivant :

<http://www.barelyfitz.com/screencast/html-training/css/positioning/>

Il existe en CSS3 un outil plus simple appelé Flexbox pour positionner les éléments dans une page Web de manière « responsive ». Les flexbox sont uniquement supporté sur les navigateurs plus récent et ne seront pas vu dans ce cours. Vous pouvez visiter le site Web suivant pour plus d'information sur les Flexbox :

- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

## Reset CSS

Chaque navigateur Web à ses propres valeurs par défaut pour certaines propriétés. Pour vous assurer que les valeurs de base sont les mêmes sur tous les navigateurs, vous pouvez utiliser un script de « reset CSS » qui mettra les mêmes valeurs de base pour tous les navigateurs. Vous en trouverez un ici :

- <http://html5doctor.com/html-5-reset-stylesheet/>

# Priorité

Plusieurs styles peuvent être affectés à un seul élément. L'ordre de base des règles CSS, appelé la cascade (CSS = Cascading Style Sheet), est défini comme suit :

1. Styles de base du navigateur
2. Styles définis dans les balises « style » ou dans les fichiers .css.
3. Styles définis par l'utilisateur qui visualise le document.
4. Styles définis dans l'attribut style des éléments HTML.

À l'intérieur de nos fichiers CSS, la priorité des règles CSS est défini par leur sélecteur. Les règles et le calcul des priorités est assez complexe. Il est tout de même intéressant de savoir que les sélecteurs par id ont une priorité sur les sélecteurs par classe. Vous trouverez plus d'information sur la page Web suivante :

- [https://developer.mozilla.org/fr/Apprendre/CSS/Introduction\\_%C3%A0\\_CSS/La\\_cascade\\_et\\_l\\_h%C3%A9ritage](https://developer.mozilla.org/fr/Apprendre/CSS/Introduction_%C3%A0_CSS/La_cascade_et_l_h%C3%A9ritage)

# Validateur CSS

Pour valider que votre CSS est conforme aux règles définies par le W3C, vous pouvez utiliser un validateur CSS. Cela assure que vos fichiers CSS seront toujours interprétés de la même façon par les navigateurs Web. Voici un validateur CSS en ligne :

- <http://www.css-validator.org/>